CS4051 23-24 Spring Coursework

General

This is the assessment component number 3 and weighs 70% of overall module mark.

Apart from the specific requirements of the Assignment Tasks, please ensure that your academic report adhere to the following guidelines to make them easier to manage, read, understand and test/modify.

- Academic Report:
  - Name your file as CS4051-studentID where Student ID is your numeric ID.
  - A template for academic report is shared in Assessments folder.
  - Similarity Report is active on Online Submission link.

- Comments for the code:
  - Start each program with a *header (comment) block* that includes:
    - Name of Program
    - Purpose of Program
    - Author of the Program
    - Date Programmed
  - **Include appropriate comments throughout the program on each line, to explain What the code does?, How the code does it? and Why you have implemented it that way?.**
- Testing
  - Make sure your program meets all the assignment requirements - *If you can't do something write out an appropriate Pseudo Code instead of the required code*.
  - Remember to test and include the test results/screenshots to your academic report.

- Required files in Coursework Submission:

  - An academic report written in MS Word or similar text editor, with full code and in-depth comments added to appendix in copy/paste-able format.

Part 1 [40 marks total]

This task will test your ability to use control flow statements and built-in functions.

Using what you have learned in this module, write a student marks calculation application to ask the user for a list of marks' of students which should be entered using the keyboard.

The numbers should be entered one at a time. You should devise a way for the user to tell the application that all the numbers have been entered. *Hint: use a loop*.

[10 marks]

The application should tell the user how many numbers they have entered and then present a menu of choices as follows:

1. Print the mean of the numbers **[6 marks]**
2. Print the median of the numbers **[6 marks]**
3. Print the mode of the numbers **[6 marks]**
4. Go back and enter a NEW set of numbers **[6 marks]**
5. Exit the application **[6 marks]**

Part 2 [30 Marks total]

Add more functionality as follows:

1. Allow only numbers to be added, give an error message to the user prompting them to enter the value again. **[10 marks]**
2. Check that at least two numbers have been added before presenting the menu. **[5 marks]**
3. Find the skewness of the values in the list **[15 marks]**

Part 3 [20 Marks total]

Add more functionality as follows:

1. Allow multiple numbers to be added as a string with the numbers separated by commas (as well as individually). *Hint: you will need to check whether a number entered contains one or more commas*. **[7 marks]**

2. Add another menu choice to enter more numbers (to the ones already entered).

[7 marks]

3. Add the functionality to read the data file from the computer hard-drive

[6 marks]

Comments in the Code [10 Marks total]

You are required to write comments within the code to explain it line by line.

IMPORTANT

Students who didn't use any comments <u>will not</u> be able to pass the coursework due to lack of evidence of understanding of concepts.

Students who used packages in Python, won't be able to get the marks for the section its used.